IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 27, 2020, accepted January 7, 2021, date of publication January 12, 2021, date of current version January 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051045

Reliability-Aware Service Function Chain Backup Protection Method

DONG ZHAI[1], XIANGRU MENG[1], ZHENHUA YU[2], (Member, IEEE), AND XIAOYANG HAN[1]

[1]College of Information and Navigation, Air Force Engineering University, Xi'an 710077, China
[2]College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an 710054, China

Corresponding author: Zhenhua Yu (zhenhua_yu@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61873277, 61401499, and 61901509; and in part by the Key Research and Development Program of Shaanxi Province under Grant Nos. 2020GY-026 and 2019GY-056.

ABSTRACT Network Function Virtualization (NFV) decouples network functions from hardware, improves the flexibility of resource allocation, and enhances network scalability. Any failures of software and hardware in an NFV environment will result in the interruption of Service Function Chains (SFCs). As one of the key technologies of 5G, NFV has more stringent delay and reliability requirements for services. This paper takes software and hardware failures into account, and proposes a reliability-aware service function chain backup protection (RABP) method to meet SFCs' high reliability and low latency demands. First, we formulate a mixed-integer linear programming model to maximize the revenue-to-cost ratio. Then, we propose a heuristic algorithm for suboptimal solutions. SFC deployment is divided into two stages: primary virtual network function (VNF) deployment and backup VNF deployment. The reliability enhancement and delay optimization deployment method is presented to deploy primary VNFs. A breadth-first search method is used to consolidate primary VNFs as much as possible to improve reliability of SFCs and reduce transmission delay. When deploying backup VNFs, a resource-efficient backup selection method is presented to reduce backup resource consumption while meeting reliability demands. Finally, experiment results show that the proposed RABP method not only has the best acceptance ratio and long-term average revenue to cost ratio, but also reduces backup resource consumption and transmission delay.

INDEX TERMS Network function virtualization, service function chain, reliability, backup, transmission delay.

I. INTRODUCTION

In recent years, network service demands have increased dramatically. For traditional networks, software and hardware are tightly coupled, and specific network functions (e.g., firewalls, intrusion detection systems and network address translation) run on dedicated hardware, which significantly increases capital and operating expenditures. Thus, it is difficult to deploy new network functions to dedicated hardware [1], [2]. As a result, traditional networks cannot meet current network business demands, leading to the phenomenon of network ossification.

Network function virtualization (NFV) is an effective way to solve network ossification. NFV decouples network functions from hardware and implements them by deploying virtual network functions (VNFs) to general servers [3], [4]. Specific VNFs are connected in a specific order to form a service function chain (SFC) [5], [6], and various services in NFV environments are implemented through SFCs. Network operators can effectively reduce the cost of providing services while meeting various business demands by using the NFV technology. A major challenge of NFV is the deployment of SFCs, including the deployment of VNFs and virtual links, which is an NP-hard problem [7].

Most current works assume that software and hardware are completely reliable, and they study the ability of efficient SFC deployments to improve the revenue of network operators. NFV deploys VNFs on commodity (e.g., x86) servers, which improves the flexibility of resource allocation and enhances the scalability of networks.

However, the vulnerability of VNF introduces significant challenges to the reliability of SFCs. The factors that lead to VNF failures are complex and diverse. For example, hardware failures associated with processor, memory, storage, and

The associate editor coordinating the review of this manuscript and approving it for publication was Xueqin Jiang.

network interface, or software failures associated with host operating systems, hypervisor, virtual machines, and VNF software configuration will cause SFC failures. Even one failure among these functional nodes can lead to the failure of the entire relevant SFCs, resulting in service interruption, data loss and waste of resources. One of the challenges is how to improve the reliability of SFCs. 5G mobile communication networks place stringent requirements on communication delays, where end-to-end demands can reach as low as a millisecond level in low-latency scenarios. Another challenge of NFV is how to effectively reduce the end-to-end delay of SFC.

There are two main ways to improve the reliability of SFCs. The first is to select a substrate node with high reliability to deploy VNF. Its drawbacks are that the reliability improvement degree is limited, and it may cause an increase of substrate link hop counts, transmission delays and greater bandwidth resource consumption. The second is to use a redundant backup method. However, it also increases resource consumption.

Some works assume that a substrate node can host more than one VNF from different SFCs, but it can host only one VNF of an SFC [8]–[12]. The reliability of SFCs is mainly improved by selecting substrate nodes having high reliability for VNF deployment. The studies in [5], [13], [14] consolidate adjacent VNFs of an SFC on the same substrate node, which can reduce the number of deployed substrate nodes, improve SFC reliability, and reduce resource consumption and transmission delay. Therefore, it is an effective way to improve the reliability of SFCs. In most works, reliability models only consider hardware failures [10]–[12], [15]. The failures of VNF may affect the reliability of SFC. And the failures of hardware may also result in the unavailability of several VNFs in one SFC. Wang *et al*. [16] take software and hardware failures into account and propose a reliability model.

Since both types of failures can cause service interruptions, this paper proposes new backup methods for software failures and a resource-efficient backup selection method considering software and server node failures.

The main contributions of this paper can be summarized as follows.

(i) We formulate a mixed-integer linear programming (MILP) model for the SFC deployment problem with high reliability and low latency demands. The objective function is to maximize the revenue-to-cost ratio. We divide SFC deployment into primary and backup VNF deployment. We take the reliability of VNFs and server nodes into account, and propose a reliability-aware service function chain backup protection (RABP) method.

(ii) We present a reliability enhancement and delay optimization (REDO) deployment method to deploy primary VNFs. It improves the reliability of SFCs, and reduces resource consumption and transmission delay by consolidating VNFs and hop constraints.

(iii) We propose new backup methods for software failures and a resource-efficient backup selection (REBS) method. The REBS method reduces backup resource consumption while satisfying reliability demand of SFCs.

The rest of this paper is organized as follows. In Section II, we discuss related works. In Section III, a network model is constructed, and a problem statement is presented. In Section IV, we propose the MILP of a reliable SFC deployment. The RABP method and its details are explained in Section V. In Section VI, we evaluate the proposed method through simulations and experiments. Section VII concludes this paper.

II. RELATED WORK

To improve the revenue of network operators, most works mainly focus on minimizing the cost of substrate network resources, while increasing the amount of business co-existence [14], [17]–[19]. Nejad *et al*. [20] solve the joint problem of admission control and SFC deployment to improve the resource utilization rate. The approach in [21] uses a feedback mechanism to deploy VNFs and virtual links, and it achieves load balancing and improves the acceptance rate. Raayatpanah and Weise [22] propose an integer linear programming to deploy VNFs and virtual links to minimize energy overhead.

To improve SFC reliability, Tang *et al*. [8] propose an SFC deployment algorithm based on queue-aware to improve the stability and reliability of services without adopting a redundant backup method. The study in [13] proposes VNF consolidation, where adjacent VNFs of an SFC could be deployed on the same substrate node to improve SFC reliability. However, the specific consolidation method is not provided.

Presently, most studies improve the reliability of SFC by adopting redundant backups. The approach in [23] provides backup resources for the entire SFC to improve its reliability, but it also increases resource consumption. To reduce backup resource consumption, the approach in [24] shares backup resources among different SFCs. Liu *et al*. [9] use a k-shortest path algorithm to deploy primary VNFs, and adopts a joint backup method during backup VNF deployment. The backup selection process is modeled as a Markov process. Backup VNFs are deployed using a Q-learning algorithm to improve the reliability of SFCs and reduce resource consumption.

While meeting SFC reliability requirements, other performance also needs to be optimized. The approach in [10] reduces bandwidth resource consumption via iterative backup and greedy shortest path algorithms. The work in [11] proposes a shortest path algorithm based on the greedy algorithm, which deploys backup VNFs using a resource sharing mechanism, whereas a forward shortest path algorithm and a backward shortest path algorithm are employed to avoid local optimization. Thus, the resource utilization ratio is improved. Qu *et al*. [15] combine iterative backup and link selection, and propose a GPS algorithm based on the greedy

algorithm, which reduces transmission delay. The approach in [12] deploys primary VNFs by a k-shortest path algorithm, and it uses a hybrid routing scheme to deploy backup VNFs. The approach effectively reduces transmission delay.

Fan *et al*. [25] propose an optimization algorithm to estimate the number of required backup VNFs in the case of heterogeneous equipment failures. Dinh and Kim [26] study the effect of sharing VNFs for SFC reliability. The approach in [27] proposes a method to determine the minimum number of required backup VNFs to ensure the reliability of SFCs against single substrate node failures. Karimzadeh-Farshbafan *et al*. [28] propose a deployment method based on the Viterbi algorithm. The method jointly deploys primary and backup VNFs to minimize resource consumption and maximize SFC reliability. The study in [29] proposes a deployment method based on the layered graphs approach, improving the reliability of SFCs and reducing transmission delay. Sun *et al*. [30] exchange the locations of function and forwarding nodes to reduce bandwidth resource consumption. The work in [31] considers the heterogeneous resource requirements of VNFs for backup VNF deployment. Aidi *et al*. [32] migrate VNF instances to reduce the number of required backup VNFs and resource consumption.

The studies in [10]–[12], [15], [33] only consider hardware failures, and the reliability of an SFC is determined by the reliability of substrate nodes, which host the VNFs of an SFC. The studies in [9], [24], [34], [35] only consider software failures, the reliability of an SFC is determined by the reliability of VNFs. Herker *et al*. [36] mention hardware failures in data center networks, but do not provide a clear reliability model that considers both VNF and hardware failures. Wang *et al*. [16] propose a reliability model accounting for software and hardware failures, and they adopt a sharable backup mechanism to reduce resource consumption.

Even if the reliability of a server node hosting the VNF is higher, the service will be interrupted if the VNF fails. The reliability of VNFs will affect the reliability of SFC and the deployment of backup VNFs.

A server node can host more than one VNF from different SFCs, but it can only host one VNF of an SFC [10]–[12], [15], [33]. The works in [5], [13], [14] assume that a server node can host more than one VNF of an SFC. They consolidate adjacent VNFs to improve the utilization of substrate network resources and reduce link transmission delay. In order to improve the success ratio of consolidation, the study in [5] sets that maximum of two VNFs can be deployed on a server node.

The works in [10]–[12], [15], [33] set that a server node can host all types of VNFs. The works in [5], [38] consider location constraints for the deployment of VNF instances and the number of licenses owned by VNF operators. They set that a server node can host several types of VNFs. The hosting capacity of VNF types will affect the deployment of VNFs.

We assume that hardware services are provided by multiple infrastructure providers. Thus, the reliability and hosting capacities of underlying general server nodes may be different. This paper takes software and server node failures into account, and assumes that a server node can host several types of VNF.

III. NETWORK MODEL AND PROBLEM STATEMENT

A. NETWORK MODEL

1) SUBSTRATE NETWORK

A substrate network (SN) is composed of a series of substrate nodes connected by substrate links. Substrate nodes include server and switch nodes. Server nodes have attributes of reliability, hosting capacity, and central processing unit (CPU) resources. Switch nodes have forwarding resource attributes. Substrate links have bandwidth resource attributes. A substrate network is modeled as a weighted undirected graph $G_p = (V_p, E_p)$, in which a substrate node set is represented by $V_p = \{v_i \,|i = 1, 2, \ldots, |V_p|\}$, and a substrate link set is represented by $E_p = \{e_i \,|i = 1, 2, \ldots, |E_p|\}$. The notations $|V_p|$ and $|E_p|$ denote the number of substrate nodes and substrate links, respectively.

A server node set is represented by $V_{p,s} = \{v_{s,i} \,|i = 1, 2, \ldots, |V_{p,s}|\}$. The notation $C(v_{s,i})$ denotes the available CPU resources of server node $v_{s,i}$. The notation $r(v_{s,i})$ denotes the reliability of server node $v_{s,i}$. The notation $|V_{p,s}|$ denotes the number of server nodes. A switch node set is represented by $V_{p,r} = \{v_{r,i} \,|i = 1, 2, \ldots, |V_{p,r}|\}$, and the notation $|V_{p,r}|$ denotes the number of switch nodes. The number of substrate nodes equals the sum of the number of server nodes and switch nodes, $|V_p| = |V_{p,s}| + |V_{p,r}|$. The notation $B(e_i)$ denotes the available bandwidth resources of substrate link e_i.

2) SFC REQUEST

Each SFC request consists of multiple VNFs connected in a specific order. VNFs have resource demands and reliability attributes. Virtual links have bandwidth demands. SFC(g) denotes the g-th SFC. It is modeled as a directed graph $G_g = \{N_g, L_g, S_g, T_g\}$, in which the VNF set is represented by $N_g = \{f_j \,|j = 1, 2, \ldots, |N_g|\}$, and the virtual link set is represented by $L_g = \{l_j \,|j = 1, 2, \ldots, |L_g|\}$. The notations S_g and T_g denote the source node and destination node of SFC(g), respectively. The notations $|N_g|$ and $|L_g|$ denote the number of VNFs and virtual links, respectively. The notation $C(f_j)$ denotes the CPU resource demand of VNF f_j. The notation $r(f_j)$ denotes the reliability of VNF f_j. The notation $f_{b,j}$ denotes the backup of VNF f_j. The notation $B(l_j)$ denotes the bandwidth resource demand of virtual link l_j. The notation $l_{b,j}$ denotes the backup of l_j. The substrate nodes where S_g and T_g are deployed are randomly determined according to the SFC(g). The notations $r(S_g)$ and $r(T_g)$ denote the reliability of substrate nodes on which S_g and T_g are deployed, respectively. The notation $R_{n,g}$ denotes the reliability demand of SFC(g).

3) SFC DEPLOYMENT

SFC deployment refers to deploying VNFs of SFC requests and virtual links between VNFs to a substrate network. Figure 1 shows the deployment of an SFC. Red and green

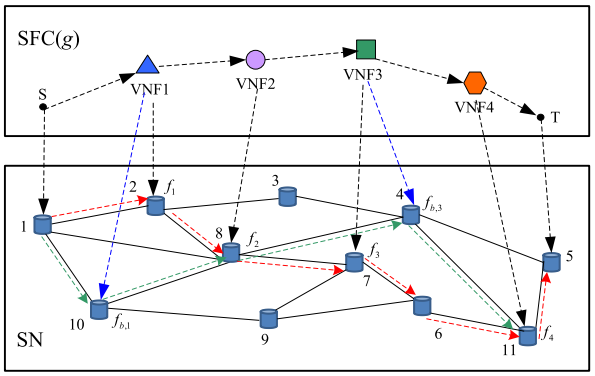

FIGURE 1. SFC(g) deployment.

dotted lines denote the deployed primary and backup links of SFC(g), respectively.

We need to consider hosting capacity attributes and resource attributes of server nodes when deploying VNFs. We need to consider bandwidth resource attributes of substrate links when deploying virtual links. SFC(g) deployed successfully can be represented by $P_g = (V_g^S, E_g^S)$, in which V_g^S and E_g^S denotes the deployed substrate node set and substrate link set, respectively.

The deployed substrate node set includes forwarding and function nodes, $V_g^S = V_{gfo}^S + V_{gf}^S$. The notation V_{gfo}^S represents a forwarding node set (e.g., node 6). The notation V_{gf}^S represents a function node set. The function nodes include the server nodes hosting primary VNFs and the server nodes hosting backup VNFs, $V_{gf}^S = V_{gfp}^S + V_{gfb}^S$. The notation V_{gfb}^S represents the server node set (e.g., nodes 4 and 10) hosting backup VNFs. The set $V_{gfp}^S = \left\{ v_{gfp,i} \middle| i = 1, 2, \ldots, \left| V_{gfp}^S \right| \right\}$ represents the server node set (e.g., nodes 2, 7, 8 and 11) hosting primary VNFs. The notation V_{gfp}^S includes the server nodes hosting the primary VNFs without backup instances and the server nodes hosting the primary VNFs with backup instances, $V_{gfp}^S = V_{gfp1}^S + V_{gfp2}^S$. The set $V_{gfp1}^S = \left\{ v_{gfp1,i} \middle| i = 1, 2, \ldots, \left| V_{gfp1}^S \right| \right\}$ represents the server node set (e.g., nodes 8 and 11) hosting the primary VNFs without backup instances. The set $V_{gfp2}^S = \left\{ v_{gfp2,i} \middle| i = 1, 2, \ldots, \left| V_{gfp2}^S \right| \right\}$ represents the server node set (e.g., nodes 2 and 7) hosting the primary VNFs with backup instances.

The notation $f_g(v_{gfp,i})$ denotes the VNF type of SFC(g) hosted by server node $v_{gfp,i}$. The server node set on which SFC(g) is not deployed is represented by $V_{ps,g} = \left\{ v_{s,j} \middle| j = 1, 2, \ldots, \left| V_{ps,g} \right| \right\}$. The notation $v_{s,j}^i$ denotes the server node that hosts the VNF f_i. If the VNF f_j of SFC(g) deployed onto service node $v_{gfp,i}$ has a backup instance, then $Ba_g^i = 1$. Otherwise, $Ba_g^i = 0$.

The notation E_g^S includes the substrate links hosting primary links and the substrate links hosting back links, $E_g^S = (E_{gp}^S + E_{gb}^S)$. The set $E_{gb}^S = \left\{ e_{b,i}^g \middle| i = 1, 2, \ldots, \left| E_{gb}^S \right| \right\}$ represents the substrate link set hosting back links. The notation E_{gp}^S represents the substrate link set hosting primary links. It includes the substrate link hosting primary links without backup and the substrate link hosting primary links with backup, $E_{gp}^S = E_p^g + E_{pb}^g$. The set $E_p^g = \left\{ e_{p,i}^g \middle| i = 1, 2, \ldots, \left| E_p^g \right| \right\}$ represents the substrate link set hosting primary links without backup. The set $E_{pb}^g = \left\{ e_{pb,i}^g \middle| i = 1, 2, \ldots, \left| E_{pb}^g \right| \right\}$ represents the substrate link set hosting primary links with backup. The notation R_g represents the reliability of SFC(g), and the notation Ti_g represents the survival time of SFC(g).

We assume that a server node can only host several types of VNFs. If server node $v_{s,i}$ can host VNF type of f_j, then $fc(v_{s,i}, f_j) = 1$. Otherwise, $fc(v_{s,i}, f_j) = 0$. If server node $v_{s,i}$ can host VNF types of f_j and f_{j+1} simultaneously, then $fc(v_{s,i}, f_j, f_{j+1}) = 1$. Otherwise, $fc(v_{s,i}, f_j, f_{j+1}) = 0$.

The main notations used in this paper are listed in Table 1

B. PROBLEM STATEMENT

This paper accounts for software and server node failures, and studies the reliability-aware SFC backup protection method. The proposed method reduces resource consumption and transmission delay, and improves the revenue to cost ratio while meeting SFC reliability demands. The reliability-aware SFC deployment is divided into two stages: primary VNF and backup VNF deployment.

Both server nodes and software have probabilities of failure. The reliability of different server nodes and types of VNFs also varies. The reliability of a server node (or a VNF) can be calculated by the mean time between failures (MTBF) and the mean time to repair (MTTR). We assume that the reliability of each server node or each VNF is independent. Thus, the reliability of server node (or VNF) can be characterized as follows:

$$r(v_{s,i})(or\ r(f_j)) = \frac{MTBF}{MTBF + MTTR} \tag{1}$$

where $r(v_{s,i})$ (or $r(f_j)$) indicates the reliability of the server node (or the VNF).

We need to deploy primary VNFs reasonably to improve SFC reliability as much as possible. This can be achieved by consolidating VNFs that deploys adjacent VNFs to the same server nodes. As shown in Figure 2, the number nearest a VNF denotes its reliability, and the two numbers nearest a server node denote its number and reliability, respectively. Figure 2(a) shows a non-consolidation state. The reliability of the SFC is 0.67, and the link hop count is five. Figure 2(b) shows a consolidation state. The reliability of the SFC is 0.71, and the link hop count is four.

Consolidating VNFs can effectively improve SFC reliability and reduce link hop counts. However, owing to restrictions or conflicts between functions, some VNFs cannot be consolidated on the same server node. This is called a function mutex constraint. If VNF f_j can be consolidated with VNF f_i, then $y(f_j, f_i) = 1$. Otherwise, $y(f_j, f_i) = 0$. For example, the multimedia resource function and mobility management entity cannot be consolidated on the same server node, because server nodes cannot provide both

TABLE 1. Notations.

Notations	Definitions
G_p	Substrate network
V_p	Set of substrate nodes
E_p	Set of substrate links
$V_{p,s}$	Set of server nodes
$V_{p,r}$	Set of switch nodes
$C(v_{s,i})$	Available CPU resources of server node $v_{s,i}$
$r(v_{s,i})$	Reliability of server node $v_{s,i}$
$B(e_i)$	Available bandwidth resources of substrate link e_i
G_g	The g-th SFC
N_g	VNF set of SFC(g)
L_g	Virtual link set of SFC(g)
S_g	Source node of SFC(g)
T_g	Destination node of SFC(g)
$C(f_j)$	CPU resource demand of VNF f_j
$r(f_j)$	Reliability of VNF f_j
$f_{b,j}$	Backup of VNF f_j
$B(l_j)$	Bandwidth resource demand of virtual link l_j
$l_{b,j}$	Backup of virtual link l_j
V_g^S	Set of substrate nodes hosting SFC(g)
E_g^S	Set of substrate links hosting SFC(g)
V_{gfo}^S	Set of forwarding nodes hosting SFC(g)
V_{gf}^S	Set of function nodes hosting SFC(g)
V_{gfp}^S	Set of function nodes hosting primary VNFs of SFC(g)
V_{gfb}^S	Set of function nodes hosting backup VNFs of SFC(g)
V_{gfp1}^S	Set of function nodes hosting primary VNFs without backup instances of SFC(g)
V_{gfp2}^S	Set of function nodes hosting primary VNFs with backup instances of SFC(g)
$V_{pv,g}$	Server node set that SFC(g) is not deployed
E_{gp}^S	Set of substrate links hosting primary virtual links of SFC(g)
E_{gb}^S	Set of substrate links hosting backup virtual links of SFC(g)
E_p^g	Set of substrate links hosting primary virtual links without backup of SFC(g)
E_{pb}^g	Set of substrate links hosting primary virtual links with backup of SFC(g)
$R_{n,g}$	Reliability demand of SFC(g)
R_g	Reliability of SFC(g)
Ti_g	Survival time of SFC(g)

functions simultaneously. Resource attributes and hosting capacity attributes of server nodes should also be considered when consolidating VNFs.

For primary VNF deployment, SFC reliability can be effectively improved by consolidating VNFs. However, the reliability improvement degree will be limited, and the SFC reliability may not satisfy requirements. Therefore, SFC reliability should be further improved by adding redundant backups. Traditional backup methods (i.e., methods 1 and 2) are shown in Figure 3. Compared with backup method 1, backup method 2 provides backup resource sharing so that

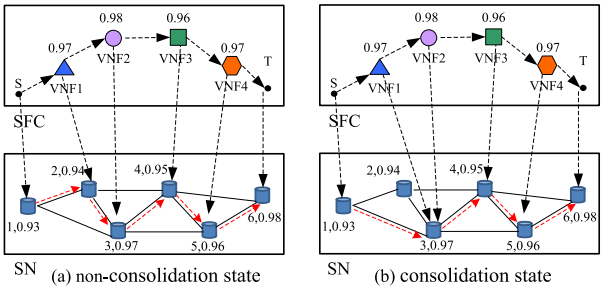

FIGURE 2. Reliability comparison between consolidation and non-consolidation states.

backup resources can be used by either of the two adjacent VNFs.

This paper proposes new backup methods for software failures. As shown in Figure 3, backup methods 3 and 4 deploy backup VNFs on the same server node, which saves more bandwidth resources than backup methods 1 and 2. Additionally, compared with backup method 3, backup method 4 provides backup resource sharing that either of the consolidated VNFs can use.

For consolidation nodes, this paper proposes backup methods 4, 5 and 6, and their reliability improvement degrees are higher than those of backup methods 3, 2 and 1, respectively. Compared with backup method 6, backup method 5 provides backup resource sharing that the VNFs hosted by a consolidation node or the VNF hosted by the adjacent server node can use. Backup methods 1, 2, 5, and 6 are helpful for software failures and server node failures, but they consume more backup resources. Backup methods 3 and 4 are only helpful for software failures, but they consume fewer backup resources. In backup VNF deployment, we select the VNF that needs to be backed up and the appropriate backup method to reduce resource consumption and maximize the revenue to cost ratio while meeting SFC reliability demands.

IV. MILP OF SFC DEPLOYMENT WITH HIGH RELIABILITY AND LOW DELAY DEMANDS

In this section, we describe the evaluation indicators and model the SFC deployment problem with high reliability and low latency demands as a mixed-integer linear programming.

A. EVALUATION INDICATORS

1) RELIABILITY OF SFCs

In this paper, we assume that substrate links and switch nodes are absolutely reliable, and we only consider the reliability of server nodes and VNFs. The reliability of SFC(g) is defined as follows:

$$R_g = r(S_g)r(T_g)\sum_{i=1}^{\left|V_{gfp1}^S\right|} R(v_{gfp1,i})\sum_{k=1}^{\left|V_{gfp2}^S\right|} R(v_{gfp2,k}) \tag{2}$$

$$R(v_{gfp1,i}) = \begin{cases} r(v_{gfp1,i}) \times r(f_j) & \text{non} - \text{consolidatednode} \\ r(v_{gfp1,i}) \times r(f_j) \times r(f_{j+1}) & \text{consolidatednode} \end{cases} \tag{3}$$

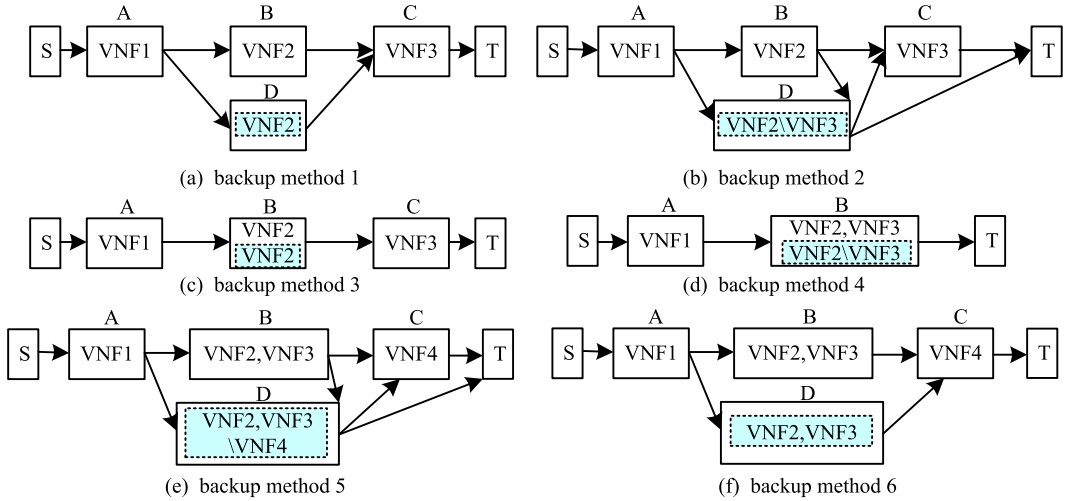

FIGURE 3. Six backup methods.

where $R(v_{gfp1,i})$ represents the overall reliability of server node $v_{gfp1,i}$ and the primary VNF without a backup instance running on server node $v_{gfp1,i}$. If server node $v_{gfp1,i}$ is a non-consolidated node, $R(v_{gfp1,i})$ is composed through the reliability of server node $v_{gfp1,i}$ and VNF f_j of the SFC(g). If the substrate node is a consolidated node, $R(v_{gfp1,i})$ is composed through the reliability of server node $v_{gfp1,i}$, and VNFs f_j and f_{j+1} of the SFC(g). $R(v_{gfp2,k})$ represents the overall reliability of the primary VNF with backup instances, its backup instances, and the server nodes on which the primary VNF and backup instances are deployed. The expression of $R(v_{gfp2,k})$ varies according to different backup methods.

For backup method 1, the expression of $R(v_{gfp2,k})$ is shown in Eq. (4).

$$\begin{aligned} R(v_{gfp2,k}) = 1 - (1 - r(v_{gfp2,k}) \times r(f_j)) \\ \times (1 - r(v_{gfpb,k}) \times r(f_j)) \end{aligned} \tag{4}$$

where $v_{gfpb,k}$ represents the backup node corresponding to $v_{gfp2,k}$, f_j is the VNF hosted by $v_{gfp2,k}$ in the SFC(g).

For backup method 2, the expression of $R(v_{gfp2,k}) \times R(v_{gfp2,k+1})$ is shown in Eq. (5).

$$\begin{aligned} &R(v_{gfp2,k}) \times R(v_{gfp2,k+1}) \\ &= r(v_{gfp2,k}) \times r(f_j) \times r(v_{gfp2,k+1}) \\ &\quad \times r(f_{j+1}) + r(v_{gfp2,k}) \times r(f_j) \times (1 - r(v_{gfp2,k+1}) \\ &\quad \times r(f_{j+1})) \\ &\quad \times r(v_{gfpb,k}) \times r(f_{j+1}) + (1 - r(v_{gfp2,k}) \times r(f_j)) \\ &\quad \times r(v_{gfp2,k+1}) \times r(f_{j+1}) \times r(v_{gfpb,k}) \times r(f_j) \end{aligned} \tag{5}$$

where $v_{gfpb,k}$ represents the shared backup node corresponding to $v_{gfp2,k}$ and $v_{gfp2,k+1}$.

For backup method 3, the expression of $R(v_{gfp2,k})$ is shown in Eq. (6).

$$R(v_{gfp2,k}) = (1 - (1 - r(f_j))^2) \times r(v_{gfp2,k}) \tag{6}$$

For backup method 4, the expression of $R(v_{gfp2,k})$ is shown in Eq. (7).

$$\begin{aligned} &R(v_{gfp2,k}) \\ &\quad = (r(f_j) \times r(f_{j+1}) + (1 - r(f_j)) \times r(f_j) \times r(f_{j+1}) \\ &\qquad + r(f_j) \times (1 - r(f_{j+1})) \times r(f_{j+1} \quad)) \times r(v_{gfp2,k}) \end{aligned} \tag{7}$$

For backup method 5, the expression of $R(v_{gfp2,k}) \times R(v_{gfp2,k+1})$ is shown in Eq. (8).

$$\begin{aligned} &R(v_{gfp2,k}) \times R(v_{gfp2,k+1}) \\ &= r(v_{gfp2,k}) \times r(f_j) \times r(f_{j+1}) \\ &\quad \times r(v_{gfp2,k+1}) \times r(f_{j+2}) + r(v_{gfp2,k}) \times r(f_j) \times r(f_{j+1}) \\ &\quad \times (1 - r(v_{gfp2,k+1}) \times r(f_{j+2})) \times r(v_{gfpb,k}) \times r(f_{j+2}) \\ &\quad + (1 - r(v_{gfp2,k}) \times r(f_j) \times r(f_{j+1})) \times r(v_{gfp2,k+1}) \\ &\quad \times r(f_{j+2}) \times r(v_{gfpb,k}) \times r(f_j) \times r(f_{j+1}) \end{aligned} \tag{8}$$

For backup method 6, the expression of $R(v_{gfp2,k})$ is shown in Eq. (9).

$$\begin{aligned} R(v_{gfp2,k}) = 1 - (1 - r(v_{gfp2,k}) \times r(f_j) \times r(f_{j+1})) \\ \times (1 - r(v_{gfpb,k}) \times r(f_j) \times r(f_{j+1})) \end{aligned} \tag{9}$$

2) RELIABILITY IMPROVEMENT TO BACKUP RESOURCE CONSUMPTION RATIO

The reliability improvement degree is defined as the reliability difference between pre-backup and after-backup SFCs, as shown in Eq. (10).

$$R_{\text{im}} = R'_g - R_g \tag{10}$$

where R'_g denotes the reliability of the after-backup SFC, and R_g denotes the reliability of the pre-backup SFC. The reliabilities of the after-backup SFC are different when adopting different backup methods. We use $R'_{g1}, R'_{g2}, R'_{g3}, R'_{g4}, R'_{g5}$ and R'_{g6} to represent the reliability of the SFC after adopting the above six backup methods in order.

The backup resource consumption is defined as increased consumption of CPU and bandwidth resources for providing

the backup, as shown in Eq. (11).

$$Res_{im} = C_b + B_b \tag{11}$$

where C_b and B_b represent increased consumptions of CPU and bandwidth resources for providing backup, respectively.

The reliability improvement degree to backup resource consumption ratio is defined as follows:

$$\eta = \frac{R_{\text{im}}}{Res_{im}} \tag{12}$$

where η represents the reliability improvement degree of unit resource.

3) ACCEPTANCE RATIO

The acceptance ratio is determined by the number of SFC requests that are deployed successfully and the total number of SFC requests, as shown in Eq. (13).

$$\omega = \lim_{T\to\infty} \frac{\sum_{t=0}^{T} \left|SFC_{deploy}(t)\right|}{\sum_{t=0}^{T} |SFC(t)| + \delta} \tag{13}$$

where δ is infinitely close to 0, $|SFC(t)|$ is the number of SFC requests at time t, and $\left|SFC_{deploy}(t)\right|$ is the number of SFC requests that are deployed successfully at time t.

4) LONG-TERM AVERAGE REVENUE TO COST RATIO

For SFC request $G_g = \{N_g, L_g, S_g, T_g\}$, we denote revenue $Re(G_g, t)$ and cost $C(G_g, t)$ as Eqs. (14) and (15) according to the works [39]–[44].

$$Re(G_g, t) = \alpha_1 \sum_{i=1}^{|N_S|} C\left(f_i\right) + \alpha_2 \sum_{j=1}^{|L_S|} B\left(l_j\right) \tag{14}$$

$$Co(G_g, t) = \alpha_1 \sum_{i=1}^{|N_S|} C\left(f_i\right) + \alpha_2 \sum_{j=1}^{|L_S|} h(l_j) B\left(l_j\right) \tag{15}$$

where α_1 and α_2 are weighting coefficients to balance CPU and bandwidth resources, respectively. Without loss of generality, we assume $\alpha_1 = \alpha_2 = 1$, indicating that the importance of CPU and bandwidth resources is similar. The notation $h(l_j)$ denotes the substrate link hop count corresponding to the virtual link l_j.

The long-term average revenue to cost ratio can be defined as follows:

$$Re/Co = \lim_{T\to\infty} \frac{\sum_{t=0}^{T} \sum_{G_g \subset SFC_{deploy}(t)} Re(G_g, t)}{\sum_{t=0}^{T} \sum_{G_g \subset SFC_{deploy}(t)} Co(G_g, t)} \tag{16}$$

where $SFC_{\text{deloy}}(t)$ is the SFC request set deployed successfully at time t.

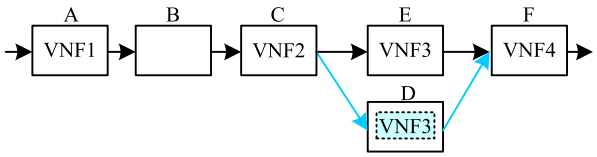

FIGURE 4. Substrate link diagram.

5) AVERAGE TRANSMISSION DELAY

The total substrate link hop counts of SFC(g) are calculated with Eq. (17).

$$H_g = \sum_{i=1}^{\left|E_p^g\right|} h(e_{p,i}^g) + \sum_{j=1}^{\left|E_{pb}^g\right|} \max(h(e_{pb,j}^g), h(e_{b,j}^g)) \tag{17}$$

where $e_{p,i}^g$ denotes the primary link deployed without backup. The notation $e_{pb,j}^g$ denotes the primary link deployed with backup. The notation $e_{b,j}^g$ denotes the backup link corresponding to $e_{pb,j}^g$. The notations $h(e_{p,i}^g)$, $h(e_{p,b,j}^g)$ and $h(e_{b,j}^g)$ indicate the hop counts of $e_{p,i}^g$, $e_{p,b,j}^g$ and $e_{b,j}^g$, respectively.

As shown in Figure 4, A, B, C, D, E and F indicate the number of substrate nodes. The link between A and C is $e_{p,i}^g$, the link between C and E is $e_{pb,j}^g$, and the link between C and D is $e_{b,j}^g$.

For convenience of analysis, this paper assumes that each hop delay of substrate links is equal, and the transmission delay of SFC(g) is defined as shown in Eq. (18).

$$Del_g = Del_0 \times H_g \tag{18}$$

where Del_0 indicates each hop delay of substrate links.

The average transmission delay is defined as follows:

$$Del\_ave = \frac{\sum_{g=1}^{NUM_{suc}} Del_g}{NUM_{suc}} \tag{19}$$

where NUM_{suc} indicates the number of SFCs that are deployed successfully.

6) AVERAGE BANDWIDTH RESOURCE CONSUMPTION

The average bandwidth resource consumption is defined as follows:

$$B\cos t\_ave = \frac{\sum_{g=1}^{NUM_{suc}} B\cos t_g}{NUM_{suc}} \tag{20}$$

where $B\cos t_g$ indicates the bandwidth resource consumption when SFC(g) is deployed successfully.

7) AVERAGE RELIABILITY

The average reliability is defined as follows:

$$R_{\text{ave}} = \frac{\sum_{g=1}^{NUM_{suc}} R_g}{NUM_{suc}} \tag{21}$$

where R_g indicates the reliability of SFC(g) when it is deployed successfully.

B. MILP

We model the SFC deployment problem with high reliability and low latency demands as a mixed-integer linear programming. The objective function and constraints can be expressed as follows.

1) OBJECTIVE FUNCTION

$$\max\left\{\lim_{T\to\infty}\frac{\sum_{t=0}^{T}\sum_{G_S\subset SFC_{deploy}(t)} Re\,(G_S,t)}{\sum_{t=0}^{T}\sum_{G_S\subset SFC_{deply}(t)} Co\,(G_S,t)}\right\} \tag{22}$$

In this paper, our object is to get the maximum long-term average revenue to cost ratio.

2) CONSTRAINTS

$$x(v_{s,i},f_j)=\begin{cases}1 & \text{if } f_j \text{ is deployed onto } v_{s,i}\\ 0 & \text{otherwise}\end{cases} \quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \tag{23}$$

$$x(v_{s,i},f_{b,j})=\begin{cases}1 & \text{if } f_{b,j} \text{ is deployed onto } v_{s,i}\\ 0 & \text{otherwise}\end{cases} \quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \tag{24}$$

$$x(e_i,l_j)=\begin{cases}1 & \text{if } l_j \text{ is deployed onto } e_i\\ 0 & \text{otherwise}\end{cases} \quad \forall e_i\in E_P,\ \forall l_j\in L_g \tag{25}$$

$$x(e_i,l_{b,j})=\begin{cases}1 & \text{if } l_{b,j} \text{ is deployed onto } e_i\\ 0 & \text{otherwise}\end{cases} \quad \forall e_i\in E_P,\ \forall l_j\in L_g \tag{26}$$

In Eq. (23), if the primary VNF f_j is deployed onto server node $v_{s,i}$, then $x(v_{s,i},f_j)=1$. Otherwise, $x(v_{s,i},f_j)=0$. In Eq. (23), if the backup VNF $f_{b,j}$ is deployed onto server node $v_{s,i}$, then $x(v_{s,i},f_{b,j})=1$. Otherwise, $x(v_{s,i},f_{b,j})=0$. In Eq. (24), if primary virtual link l_j is deployed onto substrate link e_i, then $x(e_i,l_j)=1$. Otherwise, $x(e_i,l_j)=0$. In Eq. (25), if backup virtual link $l_{b,j}$ is deployed onto substrate link e_i, then $x(e_i,l_{b,j})=1$. Otherwise, $x(e_i,l_{b,j})=0$.

$$\sum_{v_{s,i}\in V_{P,S}} x(v_{s,i},f_j)=1,\quad \forall f_j\in N_g \tag{27}$$

$$\sum_{f_j\in N_g} x(v_{s,i},f_j)\le 2,\quad \forall v_{s,i}\in V_{p,s} \tag{28}$$

$$\sum_{v_{s,i}\in V_{P,S}} x(v_{s,i},f_{b,j})\le 2,\quad \forall f_{b,j}\in N_g \tag{29}$$

$$\sum_{f_{b,j}\in N_g} x(v_{s,i},f_{b,j})\le 2,\quad \forall v_{s,i}\in V_{p,s} \tag{30}$$

Eq. (27) ensures that each VNF is deployed onto one server node. Eq. (28) ensures that each server node can host two at most VNFs of an SFC. Eq. (29) ensures that each VNF has two at most backup instances. Eq. (30) ensures that each server node can host two at most backup VNFs of an SFC.

$$x(v_{s,i},f_j,f_{j+1})=\begin{cases}1 & \text{if } f_j \text{ and } f_{j+1} \text{ are deployed onto; } v_i\\ 0 & \text{otherwise}\end{cases} \quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \tag{31}$$

In Eq. (31), if VNFs f_j and f_{j+1} are deployed onto server node $v_{s,i}$, then $x(v_{s,i},f_j,f_{j+1})=1$. Otherwise, $x(v_{s,i},f_j,f_{j+1})=0$.

$$x(v_{s,i},f_j)\times C(f_j)\le C(v_{s,i})\quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \tag{32}$$

$$x(v_{s,i},f_j,f_{j+1})\times\big(C(f_j)+C(f_{j+1})\big)\le C(v_{s,i}) \tag{33}$$

$$x(v_{s,i},f_{b,j})\times C(f_j)\le C(v_{s,i})\quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \quad \forall v_{s,i}\in V_{p,s},\ \forall f_j\in N_g \tag{34}$$

$$x(v_{s,i},f_{b,j},f_{b,j+1})\times\big(C(f_j)+C(f_{j+1})\big)\le C(v_{s,i}) \tag{35}$$

Eq. (32) ensures that a server node $v_{s,i}$ will have enough CPU resources to meet the resource demand of VNF f_j. Eq. (32) ensures that a consolidation server node $v_{s,i}$ will have enough CPU resources to meet the resource demands of VNFs f_j and f_{j+1}. Eq. (34) ensures that a backup server node $v_{s,i}$ will have enough CPU resources to meet the resource demand of backup VNF $f_{b,j}$. Eq. (35) ensures that a backup consolidation server node $v_{s,i}$ will have enough CPU resources to meet the resource demands of backup VNFs $f_{b,j}$ and $f_{b,j+1}$.

$$x(e_i,l_j)\times B(l_j)\le B(e_i)\quad \forall e_i\in E_p,\ \forall l_j\in L_g \tag{36}$$

$$x(e_i,l_{b,j})\times B(l_j)\le B(e_i)\quad \forall e_i\in E_p,\ \forall l_j\in L_g \tag{37}$$

Eq. (36) ensures that a substrate link e_i will have enough bandwidth resources to meet the bandwidth demand of virtual link l_j. Eq. (37) ensures that a backup link e_i will have enough bandwidth resources to meet the bandwidth demand of virtual link l_j.

$$e(v_i,v_j)=\begin{cases}1 & \text{if } v_i \text{ connected } v_j\\ 0 & \text{otherwise}\end{cases} \tag{38}$$

In Eq. (38), if substrate node v_i is connected to v_j, then $e(v_i,v_j)=1$. Otherwise, $e(v_i,v_j)=0$.

$$\begin{aligned}&\text{if}\quad x(v_{s,j},f_{i-1})=1,\quad x(v_{s,k},f_{i+1})=1,\quad x(v_{s,m},f_{b,i})=1\\ &\text{then}\quad e(v_{s,j},v_{s,m})=1,\quad e(v_{s,k},v_{s,m})=1,\end{aligned} \tag{39}$$

Eq. (39) denotes that the i-1th VNF f_{i-1} and $i+1$th VNF f_{i+1} of SFC(g) are deployed on server nodes $v_{s,j}$ and $v_{s,k}$, respectively. A backup instance of VNF f_i of SFC(g) is deployed on server node $v_{s,m}$. Then server node $v_{s,m}$ must be connected to server nodes $v_{s,j}$ and $v_{s,k}$. Eq. (39) ensures that the instance of backup VNF $f_{b,i}$ is connected to SFC(g).

$$h(v^i_{s,j},v^{i+1}_{s,k})\le h_0 \tag{40}$$

$$h(v^i_{s,j},v^{i+1,i+2}_{s,k})\le h_1\quad h_0<h_1 \tag{41}$$

$$h(v^{i+1}_{s,k},T)\le h(v^i_{s,j},T) \tag{42}$$

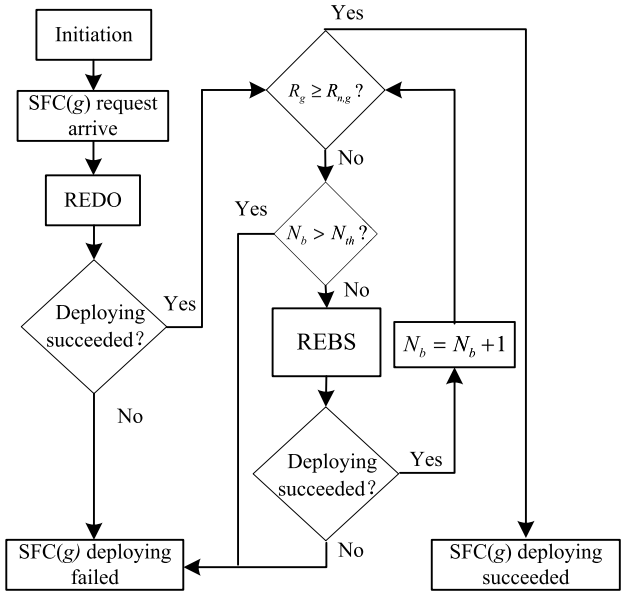

FIGURE 5. Process of RABP.

Eq. (40) ensures that hop counts between a candidate server node and the deployed adjacent server node are not greater than h_0. Eq. (41) ensures that hop counts between a candidate consolidation server node and the deployed adjacent server node are not greater than h_1. Eq. (42) ensures that hop counts between a deployed server node and the destination node do not increase. By Eqs. (40), (41) and (42), hop counts among deployed server nodes are reduced, and those from deployed server nodes to the destination node are reduced. Furthermore, transmission delay is reduced.

$$R_g \geq R_{n,g} \tag{43}$$

Eq. (43) ensures that the reliability of SFC(g) satisfies the reliability demand.

V. RABP

Based on the complexity of the above-presented MILP, the problem of finding the optimal deployment plan for one service chaining request is NP-Hard [9], [13]. Therefore, we propose the RABP method to solve it.

A. THE PROCESS OF RABP

In Figure 5, the notation N_b indicates the number of backups, and N_{th} denotes the maximum number of backups.

As shown in Figure 5, when an SFC(g) request arrives, we first deploy primary VNFs and corresponding virtual links using the REDO method to improve SFC(g) reliability and reduce transmission delay. If deployment fails, the SFC(g) request will fail. If successful, we can calculate SFC(g) reliability.

If the reliability of SFC(g) meets the reliability demand, SFC(g) deployment will succeed. Otherwise, backup VNFs will be deployed, and the VNF that needs to be backed up and the backup method will be selected using the REBS method. If the backup is successful, the reliability of SFC(g) will be updated.

Then, if the reliability of SFC(g) satisfies the reliability demand, the SFC(g) deployment will succeed. If not, judge whether the number of backups reaches the maximum number. If so, the SFC(g) deployment will fail. If not, backup VNFs will be deployed by the REBS method until the reliability demand is satisfied or the maximum number of backup is reached.

B. PRIMARY VNF DEPLOYMENT

The REDO method consolidates adjacent VNFs as much as possible, improving SFC reliability, and reducing transmission delay and resource consumption. The REDO method includes VNF deployment and virtual link deployment. VNF deployment of the REDO method is shown in **Algorithm 1**, and virtual links are deployed using the k-shortest path algorithm. For VNF deployment, a breadth-first search method is employed to judge whether the adjacent VNFs f_i and f_{i+1} of the SFC satisfy the function mutex constraint.

If $y(f_i, f_{i+1}) = 0$, the deployable candidate node set $V_1 = \{v_{s,j} | j = 1, 2, \ldots, |V_1|\}$ will be obtained by considering the resource demands of f_i, and the resource attributes and hosting capacity attributes of server nodes. If $V_1 = \phi$, the deployment will fail. If $V_1 \neq \phi$, S_1 will be taken as the original node (when f_1 is deployed, $S_1 = S$), and candidate server nodes that satisfy the hop constraint $h(S_1, v_{s,j}) \leq h_1$ will be selected in V_1 to reduce transmission delay of SFC. To avoid an increase of hop counts for the destination node, hop constraint, $h(v_{s,j}, T) \leq h(S_1, T)$, must be satisfied simultaneously. The set of candidate server nodes that satisfy hop constraints is $V_2 = \{v_{s,k} | k = 1, 2, \ldots, |V_2|\}$. If $V_2 = \phi$, the deployment will fail. If $V_2 \neq \phi$, the importance of each candidate server node will be calculated according to Eq. (49), and the server node with the largest Im value will be selected to deploy VNF f_i. The server node on which f_i is deployed is taken as S_1.

If $y(f_i, f_{i+1}) = 1$, the resource demands of f_i and f_{i+1} should be satisfied simultaneously. To improve the success rate of consolidation, the hop constraint is set as $h(S_1, v_{s,j}) \leq h_2$. The remaining steps are the same as the case of $y(f_i, f_{i+1}) = 0$.

The node degree is defined as the number of nodes connected to it, as shown in Eq. (44).

$$G(i) = N(v_i) \tag{44}$$

The node degree represents the possibility of connecting with other nodes.

When $V_2 \neq \phi$ or $V_2' \neq \phi$, for all nodes in V_2 or V_2', we normalize their available CPU resources and node degrees respectively, as shown in Eqs. (45), (46), (47) and (48).

$$C_{total} = \sum_{k=1}^{K} C(v_k) \quad K = |V_2| \, or \, |V_2'| \tag{45}$$

$$C_{nor}(v_k) = C(v_k)/C_{total} \tag{46}$$

$$G_{\text{total}} = \sum_{k=1}^{K} G(k) \quad K = |V_2| \text{ or } |V_2'| \tag{47}$$

$$G_{nor}(k) = G(k)/G_{total} \tag{48}$$

Algorithm 1 deployment process of VNF in the REDO method

Input: substrate network G_p, an SFC request G_g
Output: VNF deployment list
(1) the number of VNFs$|N_g|$
(2) for $i = 1$: $|N_g|$-1
(3) judge whether the adjacent VNFs f_i and f_{i+1} satisfy the function mutex constraint
(4) if $y(f_i, f_{i+1}) = 0$
(5) obtain candidate node sets V_1 and V_2 by Algorithm 2 to optimize transmission delay
(6) if $V_1 = \phi$
(7) deployment fails
(8) return
(9) else
(10) if $V_2 = \phi$
(11) deployment fails
(12) return
(13) else
(14) calculate *Im* of server nodes in V_2, select the server node with the largest *Im* value to deploy f_i
(15) end
(16) end
(17) else
(18) obtain candidate node sets V_1' and V_2' by Algorithm 3 to optimize reliability and transmission delay
(19) if $V_1' = \phi$
(20) consolidation fails
(21) return to step (5)
(22) else
(23) if $V_2' = \phi$
(23) consolidation fails
(24) return to step (5)
(25) else
(27) calculate *Im* of service nodes in V_2', select the server node with the largest *Im* value to deploy f_i and f_{i+1}
(28) end
(29) end
(30) end

The evaluation metric of node importance is shown in Eq. (49):

$$Im_k = \frac{\beta_1 \times C_{nor}(v_k) + \beta_2 \times G_{nor}(k)}{h(S_1, v_k)} \tag{49}$$

where β_1 and β_2 are weighting coefficients used to balance the CPU resource and node degree. Without loss of generality, we assume $\beta_1 = \beta_2 = 1$.

The deployment process of VNF in the REDO method is shown in **Algorithm 1**.

The process of obtaining candidate server node sets V_1 and V_2 is shown in **Algorithm 2**.

The process of obtaining candidate consolidation node sets V_1' and V_2' is shown in **Algorithm 3**.

Algorithm 2 process of obtaining candidate server node sets V_1 and V_2

Input: $\forall f_i \in N_g$, $V_{ps,g} = \{v_{s,j} | j = 1, 2, \ldots, |V_{ps,g}|\}$
Output: V_1, V_2
(1) for $j = 1$: $|V_{ps,g}|$
(2) if $C(v_{s,j}) \geq C(f_i) \&\& fc(v_{s,j}, f_i) = 1$
(3) $v_{s,j} \in V_1$
(4) end
(5) end
(6) for $k = 1$: $|V_1|$
(7) if $h(S_1, v_{s,k}) \leq h_1 \&\& h(v_{s,k}, T) \leq h(S_1, T)$
(8) $v_{s,k} \in V_2$
(9) end
(10) end

Algorithm 3 process of obtaining candidate consolidation node sets V_1' and V_2'

Input: $\forall f_i \in N_g$, $V_{ps,g} = \{v_{s,j} | j = 1, 2, \ldots, |V_{ps,g}|\}$
Output: V_1', V_2'
(1) for $j = 1$: $|V_{ps,g}|$
(2) if $C(v_{s,j}) \geq (C(f_i) + C(f_{i+1})) \&\& fc(v_{s,j}, f_i, f_{i+1}) = 1$
(3) $v_{s,j} \in V_1'$
(4) end
(5) end
(6) for $k = 1$: $|V_1'|$
(7) if $h(S_1, v_{s,k}) \leq h_2 \&\& h(v_{s,k}, T) \leq h(S_1, T)$
(8) $v_{s,k} \in V_2'$
(9) end
(10) end

C. BACKUP VNF DEPLOYMENT

In this paper, we propose the REBS method, which selects the VNF that needs to be backed up by the overall reliability of server nodes and their deployed VNFs of SFC(g). We select the server node with the lowest overall reliability and provide it with a backup for the deployed VNF of SFC(g). Considering the reliability of SFCs, the reliability improvement degree to backup resource consumption ratios by adopting different backup methods, and the reliability demands, we select a resource-efficient backup method to minimize backup resource consumption. Backup virtual links are deployed using the k-shortest path algorithm.

The REBS method is shown in **Algorithm 4**. For SFC(g), we calculate the overall reliability of deployed server nodes, and select the server node with the lowest overall reliability and provide it with a backup for deployed VNF in SFC(g). During backing up, judge whether the server node is a consolidation node. If $x(v_{s,i}, f_j, f_{j+1}) = 1$, the server node is a consolidation node, and a resource-efficient backup method will be selected among backup methods 3, 4, 5 and 6 (Lines 5–25). If $x(v_i, f_j, f_{j+1}) = 0$, the server node is a non-consolidation node, and a resource-efficient backup method will be selected among backup methods 1, 2 and 3 (Lines 30–42). We calculate the reliability of SFC after adopting the

Algorithm 4 REBS

Input: $V^S_{gfp1} = \left\{ v_{gfp1,i} \middle| i = 1, 2, \ldots, \left| V^S_{gfp1} \right| \right\}$
Output: backup method
(1) for i=1: $\left| V^S_{gfp1} \right|$
(2) calculate $R(v_{gfp1,i})$
(3) end
(4) select the server node with the lowest overall reliability $R(v_{gfp1,i})$ and provide it with a backup for deployed VNF in SFC(g)
(5) if $x(v_{s,i}, f_j, f_{j+1}) = 1$
(6) if $C(v_{s,i}) \geq max\left\{C(f_j), C(f_{j+1})\right\}$
(7) calculate R'_{g3} and R'_{g4}
(8) else
(9) if $C(v_{s,i}) < max\left\{C(f_j), C(f_{j+1})\right\}$ &&$C(v_{s,i}) \geq mix\left\{C(f_j), C(f_{j+1})\right\}$
(10) calculate R'_{g3} and set $R'_{g4} = 0$
(11) else (12) set $R'_{g3} = 0$ and $R'_{g4} = 0$
(13) end (14) end
(15) if $R'_{g3} \geq R_{n,g}$
(16) adopt backup method 3
(17) else
(18) if $R'_{g4} \geq R_{n,g}$&&$R'_{g3} < R_{n,g}$
(19) adopt backup method 4
(20) else
(21) calculateR'_{g5} and R'_{g6} by Algorithm 5
(22) if $R'_{g5} \geq R_{n,g}$ or $R'_{g6} \geq R_{n,g}$
(23) select the backup method that satisfies the reliability demand and has the least backup resource consumption
(23) else
(24) calculate η for the above four backup methods, and select the backup method with the maximum η
(25) end
(27) end
(28) end
(29) else
(30) if $C(v_{s,i}) \geq C(f_j)$
(31) calculate R'_{g3}
(32) else
(33) set $R'_{g3} = 0$
(34) end
(35) if $R'_{g3} \geq R_{n,g}$
(36) adopt backup method 3
(37) else
(38) calculateR'_{g1} and R'_{g2} by Algorithm 6
(39) if $R'_{g1} \geq R_{n,g}$ or $R'_{g2} \geq R_{n,g}$
(40) select the backup method that satisfies the reliability demand and has the least backup resource consumption
(41) else
(42) calculate η for the above three backup methods, and select the backup method with the maximum η
(43) end
(44) end
(45) end

above backup methods, and select the backup method that meets the reliability demand and has the minimum resource consumption. If none of the above backup methods meet the reliability demand, we will adopt the backup method with the largest reliability improvement to resource consumption ratio.

The process of calculating the reliability of SFC by adopting backup methods 5 and 6 is shown in **Algorithm 5**. For backup method 6, to avoid the increase of bandwidth resource consumption caused by the increase of link hops, $v_{gfp,i-1}$ is taken as the origin node, and candidate server node ω_4 is obtained by hop constraints and hosting capacity attributes of server nodes in the server node set $V_{ps,g} = \left\{ v_{s,k} \middle| k = 1, 2, \ldots, \left| V_{ps,g} \right| \right\}$ (Lines 2–9). We select the server node having the smallest hop count with $v_{gfp,i-1}$ and $v_{gfp,i+1}$ as the backup node in ω_4, and we calculate R'_{g6}.

Method 5 can be further divided into two backup methods. One is to share backup resources with the previous adjacent deployed server node, and the other is to share backup resources with the next adjacent deployed server node. These two methods are referred to as backup methods 5.1 and 5.2, respectively. To maximize the SFC reliability improvement degree of a unit resource, if adjacent nodes have backups, backup resources will no longer be shared with them. The reliabilities of backup methods 5.1 and 5.2 are then calculated (Lines 13–30). We also introduce the expression, $R'_{g5} = \max\left\{R'_{g5.1}, R'_{g5.2}\right\}$.

The process of calculating SFC reliability by adopting backup methods 1 and 2 is shown in **Algorithm 6**. For backup method 1, to avoid the increase of bandwidth resource consumption caused by the increase of link hops, $v_{gfp,i-1}$ and $v_{gfp,i+1}$ are taken as the original node, respectively. The candidate server node ω_{12} is obtained by hop constraints and hosting capacity attributes of server nodes in the server node set, $V_{psg} = \left\{ v_{s,k} \middle| k = 1, 2, \ldots, \left| V_{psg} \right| \right\}$ (Lines 2–9). We select the server node having the smallest hop counts with $v_{gfp,i-1}$ and $v_{gfp,i+1}$ as the backup node in ω_{12}, and we calculate R'_{g1}.

Method 2 can be further divided into two backup methods. The first is to share backup resources with the previous adjacent deployed server node, and the second is to share backup resources with the next adjacent deployed server node. The above two backup methods are referred to as backup methods 2.1 and 2.2, respectively. The reliabilities of backup method 2.1 and 2.2 are then calculated, respectively (Lines 13–32). We also introduce the expression, $R'_{g2} = \max\left\{R'_{g2.1}, R'_{g2.2}\right\}$.

D. COMPLEXITY ANALYSIS

In primary VNF deployment, the complexity of selecting candidate nodes is $O(\left|V_{p,s}\right|^2)$, and the complexity of deploying virtual links by the k-shortest path algorithm is $O(k\left|V_p\right|(\left|E_p\right| + \left|V_p\right| \lg \left|V_p\right|))$. In backup VNF deployment, the complexity of selecting candidate nodes is $O(\left|V_{p,s}\right|^2)$,

Algorithm 5 process of calculating the reliability of SFCs by adopting backup methods 5 and 6

Input: the deployed server node set V_{gf}^{S}, the server node $v_{gfp1,i}$ with lowest overall reliability, the server node set $V_{ps,g} = \left\{v_{s,j} \middle| j = 1, 2, \ldots, \left|V_{ps,g}\right|\right\}$ that VNFs of SFC(g) are not deployed

Output: R'_{g5}, R'_{g6}

(1) for $k = 1$: $\left|V_{ps,g}\right|$
(2) if $h(v_{gfp,i-1}, v_{s,k}) \leq h_3$
(3) $v_{s,k} \in \omega_1$ (4) end
(5) if $h(v_{gfp,i+1}, v_{s,k}) \leq h_3$
(6) $v_{s,k} \in \omega_2$
(7) end
(8) $\omega_3 = \omega_1 \cap \omega_2$
(9) select server node set ω_4 that satisfies the resource demand $\left(C\left(f_j\right) + C\left(f_{j+1}\right)\right)$ and the hosting capacity constraint $fc(v_{s,k}, f_j, f_{j+1}) = 1$ in ω_3
(10) select the node that has the smallest hop counts with $v_{gfp,i-1}$ and $v_{gfp,i+1}$ as the backup node in ω_4, and calculate R'_6
(11) end
(12) for $k = 1$: $\left|V_{ps,g}\right|$ (13) if $Ba_g^{i-1} = 0 \&\& Ba_g^{i+1} = 1$
(14) $R'_{5.2} = 0$
(15) if $h(v_{gfp,i-2}, v_{s,k}) \leq h_4 \&\& fc(v_{s,k}, f_g(v_{gfp1,i-1})) = 1$
(16) $v_{s,k} \in \omega_5$
(17) $\omega_6 = \omega_5 \cap \omega_4$
(18) select the node that has the smallest hop counts with $v_{gfp,i-2}$, $v_{gfp,i-1}$ and $v_{gfp,i+1}$ as the backup node in ω_6, and calculate $R'_{g5.1}$
(19) end
(20) else
(21) if $Ba_g^{i-1} = 1 \&\& Ba_g^{i+1} = 0$
(22) $R'_{g5.1} = 0$
(23) if $h(v_{gfp,i+2}, v_{s,k}) \leq h_4 \&\& fc(v_{s,k}, f_g(v_{gfp1,i+1})) = 1$
(23) $v_{s,k} \in \omega_7$
(24) $\omega_8 = \omega_7 \cap \omega_4$
(25) select the node that has the smallest hop counts with $v_{gfp,i-1}$, $v_{gfp,i+1}$ and $v_{gfp,i+2}$ as the backup node in ω_8, and calculate $R'_{g5.2}$
(27) end
(28) else
(29) return to step (15)$\sim$ (18), get $R'_{g5.1}$
(30) return to step (23)$\sim$ (25), get $R'_{g5.2}$
(31) end
(32) $R'_{g5} = \max\left\{R'_{g5.1}, R'_{g5.2}\right\}$
(32) end
(34) end

and the complexity of deploying virtual links by the k-shortest path algorithm is $O(k\left|V_p\right|(\left|E_p\right| + \left|V_p\right|\lg\left|V_p\right|))$.

Algorithm 6 process of calculating the reliability of SFC by adopting backup methods 1 and 2

Input: the deployed server node set V_{gf}^{S}, the serve node $v_{gfp1,i}$ with lowest overall reliability, the server node set $V_{ps,g} = \left\{v_{s,j} \middle| j = 1, 2, \ldots, \left|V_{ps,g}\right|\right\}$ that VNFs of SFC(g) are not deployed

Output: R'_{g1}, R'_{g2}

(1) for $k = 1$: $\left|V_{ps,g}\right|$
(2) if $h(v_{gfp,i-1}, v_{s,k}) \leq h_5$
(3) $v_{s,k} \in \omega_9$ (4) end
(5) if $h(v_{gfp,i+1}, v_{s,k}) \leq h_5$
(6) $v_{s,k} \in \omega_{10}$
(7) end
(8) $\omega_{11} = \omega_{10} \cap \omega_9$
(9) select server node set ω_{12} that satisfies the resource demand $C\left(f_j\right)$ and the hosting capacity constraint $fc(v_{s,k}, f_j) = 1$ in ω_{11}
(10) select the node that has the smallest hop counts with $v_{gfp,i-1}$ and $v_{gfp,i+1}$ as the backup node in ω_{12}, calculate R'_{g1}
(11) end
(12) for $k = 1$: $\left|V_{ps,g}\right|$
(13) if $Ba_g^{i-1} = 0 \&\& Ba_g^{i+1} = 1$
(14) $R'_{g2.2} = 0$
(15) if $h(v_{gfp,i-2}, v_{s,k}) \leq h_6 \&\& fc(v_{s,k}, f_g(v_{gfp1,i-1})) = 1$
(16) $v_{s,k} \in \omega_{13}$
(17) $\omega_{14} = \omega_{13} \cap \omega_{12}$
(18) obtain server node set ω_{15} that satisfies the resource demand max $\left\{C(f_{j-1}), C(f_j)\right\}$ in ω_{14}
(19) select the node that has the smallest hop counts with $v_{gfp,i-2}$, $v_{gfp,i-1}$ and $v_{gfp,i+1}$ as the backup node in ω_{15}, calculate $R'_{g2.1}$
(20) end
(21) else
(22) if $Ba_g^{i-1} = 1 \&\& Ba_g^{i+1} = 0$
(23) $R'_{g2.1} = 0$
(23) if $h(v_{gfp,i+2}, v_{s,k}) \leq h_6 \&\& fc(v_{s,k}, f_g(v_{gfp1,i+1})) = 1$
(24) $v_{s,k} \in \omega_{16}$
(25) $\omega_{17} = \omega_{16} \cap \omega_{12}$
(27) select serve node set ω_{18} that satisfies the resource demand max $\left\{C(f_j), C(f_{j+1})\right\}$ in ω_{17}
(28) select the node that has the smallest hop counts with $v_{gfp,i-1}$, $v_{gfp,i+1}$ and $v_{gfp,i+2}$ as the backup node in ω_{18}, calculate $R'_{g2.2}$
(29) end
(30) else
(31) return to step (15)$\sim$(19), get $R'_{g2.1}$
(32) return to step (23)$\sim$(28), get $R'_{g2.2}$
(32) end
(34) $R'_{g2} = \max\left\{R'_{g2.1}, R'_{g2.2}\right\}$
(35) end
(36) end

The complexity of the RABP method is $O(|V_{p,s}|^2 + k|V_p|(|E_p| + |V_p|\lg|V_p|))$.

VI. SIMULATION

In this paper, MATLAB is used for simulation and the RABP method proposed in this paper is evaluated in a large-scale network scenario and compared with two other methods.

A. SIMULAITON ENVIRONMENT

The substrate network topology and SFC topology used in the simulation experiments are generated by the improved Salam network topology random generation algorithm. We set the substrate switch and server nodes to be deployed to the same location of operator networks. Their numbers are both 100, and the link connection probability between the switch nodes is 0.5. The computing resources of server nodes obey the uniform distribution of [60, 100], and the link bandwidth between the substrate network switch nodes obeys the uniform distribution of [60, 100]. The reliability of the substrate network switch nodes obeys the uniform distribution of [0.96, 0.98]. It is assumed that each server node can host any two types of $\{f_1, f_2, f_3, f_4, f_5\}$.

Each SFC is composed of five VNFs. The server nodes that host source and destination nodes are randomly determined according to an SFC request. It is assumed that there are five types of VNFs$\{f_1, f_2, f_3, f_4, f_5\}$, where f_3 and f_4 cannot be consolidated. The reliability of VNFs obeys the uniform distribution of [0.97, 0.99]. The computing resource demands of VNFs obey the uniform distribution of [8], 12], and the bandwidth demands of SFCs obey the uniform distribution of [21], [24]. The arrival ratio of SFC requests obeys the Poisson distribution with the parameter 0.05 and the life time obeys the exponential distribution with parameter 1000.

The duration of simulation experiments is 10,000 time units. We set the hop constraints $h_1 = 2$, $h_2 = h_5 = h_6 = 3$ and $h_3 = h_4 = 4$. The transmission delay of each hop is set to 1ms. The maximum number of backups is 2. The reliability demand of SFCs is 0.9. To eliminate the effect of random factors on the experimental results, the simulation experiment is conducted 10 times, and their average value is taken as the final simulation results.

B. COMPARISON METHOD

To evaluate the performance of the proposed method, we compare it with the DRH-FD-Greedy [23] method and the BCR method [19] in the same experimental environment. The description of the three methods is shown in Table 2.

The delay constraint in the DRH-FD-Greedy method is set to 8ms. For a better comparison, the DRH-FD-Greedy method and BCR method are modified. The function mutex constraint is considered for the BCR method. Hosting capacity attributes of server nodes, and reliability of VNFs and substrate nodes are considered for the DRH-FD-Greedy and BCR methods.

C. EXPERIMENTAL ANALYSIS

Figure 6 illustrates the acceptance ratios of the three methods in the stable state.

TABLE 2. Description of the three mehtods.

Method	Description
RABP	Reliability-aware service function chain backup protection method proposed in this paper. In primary VNF deployment, VNFs are deployed using the REDO method. For backup VNF deployment, we select the VNF that needs to be backed up and deploy the backup VNF using the REBS method.
DRH-FD-Greedy	Reliability-aware service function chain deployment method with function decomposition and multipath routing proposed in [12]. The method only considers hardware failures. In primary VNF deployment, VNFs are deployed using the k-shortest path algorithm, function decomposition and multipath routing. For backup VNF deployment, backup VNFs are deployed by function decomposition and multipath routing.
BCR	Reliability-aware service chaining deployment method with Q-learning proposed in [9]. The method only considers software failures. In primary VNF deployment, VNFs are deployed using the k-shortest path algorithm. For backup VNF deployment, backup VNFs are deployed using the joint backup method and Q-learning algorithm.

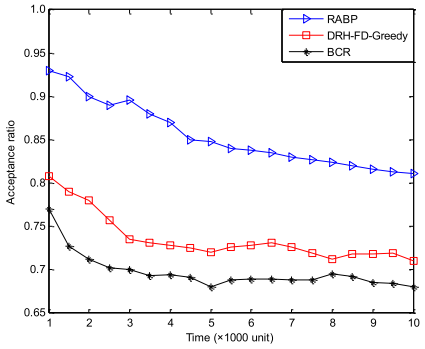

FIGURE 6. Acceptance ratios.

In the BCR method, the k-shortest path algorithm is used to deploy primary VNFs. The hop counts of substrate links and bandwidth resource consumption may increase, owing to hosting capacity attributes of substrate nodes. The joint backup method is employed to deploy backup VNFs. Backup resources are provided for two adjacent VNFs simultaneously, increasing the cost of CPU and bandwidth resources. Owing to the function mutex constraint, the two adjacent VNFs may not be backed up on the same substrate node, and the joint backup may fail. The acceptance ratio of the BCR method is close to 0.68, which is lower than those of the other two methods.

In the DRH-FD-Greedy method, the k-shortest path algorithm is employed to deploy primary VNFs, which may fail, owing to the delay constraint and hosting capacity attributes. Therefore, function decomposition and multipath routing are employed to deploy primary VNFs, which increase bandwidth resource consumption. We provide backup for the VNF having the lowest reliability so that the reliability improvement degree of unit resource would be higher than that of

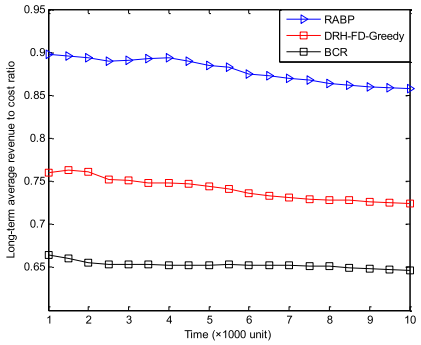

FIGURE 7. Long-term average revenue to cost ratios.

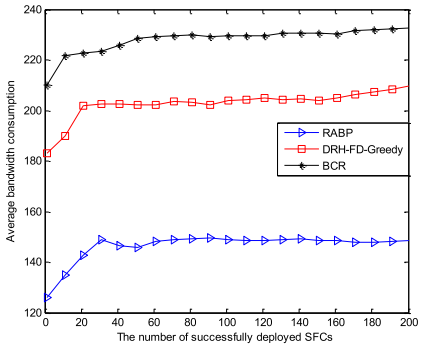

FIGURE 8. Average bandwidth consumptions.

the BCR method. The backup resource consumption of the DRH-FD-Greedy method is less than that of the BCR method. The acceptance ratio of the DRH-FD-Greedy method is close to 0.71, which is higher than that of the BCR method.

In the RABP method, primary VNFs are deployed by the REDO method, and VNFs are consolidated by a breadth-first search method as much as possible, which reduces the number of deployed server nodes and bandwidth resource consumption, and improves the reliability of SFCs. The SFC reliability that needs to be improved by backup is lower than that of the other two methods. Backup VNFs are deployed by the REBS method. The REBS method selects a resource-efficient backup method by the reliability of SFCs adopting different backup methods, reliability demand, and the η metric. The REBS method reduces the cost of computation and bandwidth resources. The resource consumption of the RABP method is the lowest among the three methods, and its acceptance ratio is close to 0.81, which is the highest among the three methods.

Figure 7 illustrates the long-term average revenue to cost ratios of the three methods in the stable state. For the BCR method, the joint backup method is adopted to deploy backup VNFs, and the reliability of adjacent VNFs may not be the lowest. The reliability improvement degree of unit resource is lower than that of the other two methods. Providing backup for two VNFs simultaneously can create unnecessary backups. The long-term average revenue to cost ratio of the BCR method is close to 0.65, which is the lowest among the three methods.

For the DRH-FD-Greedy method, backup resources are provided for the VNF with the lowest reliability, and its reliability improvement degree of unit resource is higher than that of the BCR method. The long-term average revenue to cost ratio of the DRH-FD-Greedy method is close to 0.72, which is higher than that of the BCR method. For the BCR and DRH-FD-Greedy methods, primary VNFs are deployed by the k-shortest path algorithm. The hop counts of substrate link and bandwidth resource consumption may increase because of hosting capacity attributes of the substrate nodes.

In the RABP method, when deploying primary VNFs, bandwidth resource consumption is reduced by consolidating VNFs and hop constraints. The REBS method is adopted to deploy backup VNFs. We first judge whether there are corresponding backup nodes for the six backup methods. If so, the reliability of the SFC that adopts the corresponding backup method will be calculated. If there are backup methods that meet the reliability demand, we will adopt the backup method that satisfies the reliability demand and has the least resource consumption. Otherwise, the backup method having the largest reliability improvement degree to backup resource consumption ratio is adopted. The backup VNFs are deployed using the REBS method, and backup resource consumption is reduced. The resource consumption of the RABP method is the lowest among the three methods, and its long-term average revenue to cost ratio is close to 0.85, which is the highest of the three methods.

Figure 8 illustrates the average bandwidth consumptions of the three methods in the stable state. In the BCR method, the k-shortest path algorithm is employed to deploy primary VNFs, and the hop counts of substrate links and bandwidth resource consumption will increase because of hosting capacity attributes of the substrate nodes. Backup VNFs are deployed by the joint backup method. The reliability improvement degree of the BCR method is lower than those of the other two methods, and its bandwidth resource consumption would increase. The average bandwidth consumption of the BCR method is close to 232, which is the highest among the three methods.

For the DRH-FD-Greedy method, backup resources are provided for the VNF having the lowest reliability. The reliability improvement degree of the DRH-FD-Greedy method is higher than that of the BCR method, and its backup bandwidth resource consumption is less than that of the BCR method. The average bandwidth consumption of the DRH-FD-Greedy method is close to 208, which is lower than that of the BCR method.

In the RABP method, bandwidth resource consumption is reduced by consolidating VNFs and hop constraints. Consolidating VNFs improves reliability of SFCs, and reduces the reliability of SFCs that needs to be improved by backup and the required backup bandwidth resources. Backup methods 3 and 4 proposed in this paper effectively reduce backup bandwidth resource consumption. The average bandwidth consumption of the RABP method is close to 148, which is the lowest among the three methods.

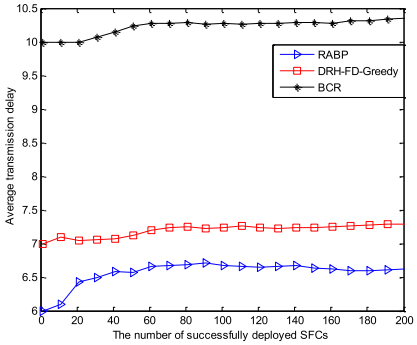

FIGURE 9. Average transmission delays.

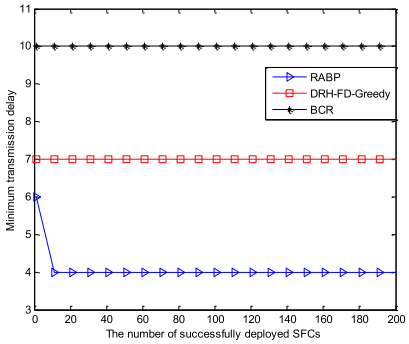

FIGURE 10. Minimum transmission delays.

Figure 9 illustrates the average transmission delays of the three methods in the stable state. Figure 10 illustrates the minimum transmission delays of the three methods in the stable state. In the BCR method, primary VNFs are deployed using the k-shortest path algorithm. The hop counts of substrate links and transmission delay will increase because of hosting capacity attributes of substrate nodes.

For the DRH-FD-Greedy method, transmission delay is reduced using the k-shortest path algorithm, function decomposition and multipath routing. The average and minimum transmission delays of the DRH-FD-Greedy method are lower than those of the BCR method.

For the RABP method, primary VNFs are deployed using the REDO method. The transmission delay is reduced by consolidating VNFs and hop constraints. Backup VNFs are deployed by multiple backup methods, and backup methods 3 and 4 effectively reduce transmission delay. The average and minimum transmission delays of the RABP method are the lowest among the three methods. It can be seen from Figures 9 and 10 that the RABP method can effectively reduce transmission delay.

Figure 11 illustrates the average backup resource consumption of the three methods in the stable state. In the BCR method, the joint backup method is employed to deploy backup VNFs. The reliability of adjacent VNFs may be not the lowest, and the reliability improvement degree of unit resource is lower than those of the other two methods. It can be observed that providing backup for two VNFs simultaneously can increase unnecessary backups. The BCR method increases backup resource consumption.

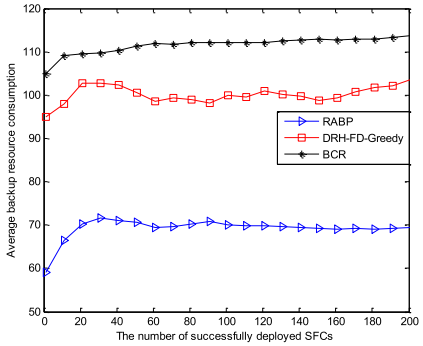

FIGURE 11. Average backup resource consumption.

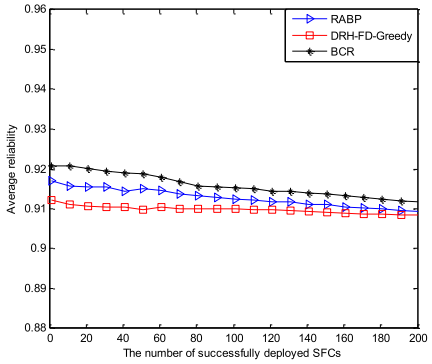

FIGURE 12. Average reliability.

For the DRH-FD-Greedy method, backup resources are provided for the VNF having the lowest reliability, and its reliability improvement degree of unit resource is higher than that of the BCR method. The average backup resource consumption of the DRH-FD-Greedy method is close to 102, which is lower than that of the BCR method.

For the RABP method, when deploying primary VNFs, reliability of SFCs is improved by consolidating VNFs, and the reliability of SFCs that needs to be improved by backup is lower than those of the other two methods. When deploying backup VNFs, backup methods 3 and 4 proposed in this paper effectively reduce backup bandwidth resource consumption. The REBS method effectively reduces backup resource consumption by adopting the backup method having the largest reliability improvement degree to backup resource consumption ratio. The average backup resource consumption of the RABP method is close to 69, which is the lowest among the three methods.

Figure 12 illustrates the average reliability of the three methods in the stable state. The BCR method provides backup resources for two adjacent VNFs simultaneously. Consequently, its average reliability is the highest among the three methods. The BCR method may provide unnecessary backup resources, and its resource consumption is the highest among the three methods. For the RABP method, if primary VNFs are deployed, the reliability of SFCs is improved by consolidating VNFs. For backup VNF deployment, providing sharing backup resources can improve the reliability of SFCs. The reliability of the RABP method is higher than that of

the DRH-FD-Greedy method. The RABP method provides necessary backup resources to satisfy reliability demands as soon as possible. Therefore its reliability is lower than that of the BCR method, and its resource consumption is the lowest among the three methods.

From Figures 6, 7, 8, 11 and 12, we can conclude that the RABP method is resource-efficient.

VII. CONCLUSION

This paper proposes a reliability-aware SFC backup protection method that divides SFC deployment into two stages: primary VNF and backup VNF deployment. The REDO method is adopted to deploy primary VNFs. It improves the reliability of SFCs, and reduces the transmission delay and bandwidth resource consumption by consolidating VNFs and hop constraints. The REBS method is adopted to deploy backup VNFs. The REBS method selects a resource-efficient backup method by the reliability of SFCs adopting different backup methods, reliability demand, and the η index, which reduces the backup resource consumption while meeting reliability demands. The simulation results show that the proposed method performs better than the other two methods in terms of acceptance ratio, long-term average revenue to cost ratio, average bandwidth consumption, average transmission delay, minimum transmission delay, and average backup resource consumption. In the future, we can study NFV from the perspective of multi-agent systems [45].

REFERENCES

[1] B. Yi, X. Wang, K. Li, S. K. Das, and M. Huang, "A comprehensive survey of network function virtualization," *Comput. Netw.*, vol. 133, no. 14, pp. 212–262, Mar. 2018.

[2] G. Sun, Z. Xu, H. Yu, X. Chen, V. Chang, and A. V. Vasilakos, "Low-latency and resource-efficient service function chaining orchestration in network function virtualization," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5760–5772, Jul. 2020.

[3] X. Chen, W. Ni, I. B. Collings, X. Wang, and S. Xu, "Automated function placement and online optimization of network functions virtualization," *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 1225–1237, Feb. 2019.

[4] M. Otokura, K. Leibnitz, Y. Koizumi, D. Kominami, T. Shimokawa, and M. Murata, "Evolvable virtual network function placement method: Mechanism and performance evaluation," *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 1, pp. 27–39, Mar. 2019.

[5] X. Han, X. Meng, Z. Yu, Q. Kang, and Y. Zhao, "A service function chain deployment method based on network flow theory for load balance in operator networks," *IEEE Access*, vol. 8, pp. 93187–93199, May 2020.

[6] G. Sun, R. Zhou, J. Sun, H. Yu, and A. V. Vasilakos, "Energy-efficient provisioning for service function chains to support delay-sensitive applications in network function virtualization," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6116–6131, Jul. 2020.

[7] B. Tahmasebi, M. A. Maddah-Ali, S. Parsaeefard, and B. H. Khalaj, "Optimum transmission delay for function computation in NFV-based networks: The role of network coding and redundant computing," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2233–2245, Oct. 2018.

[8] L. Tang, G. Zhao, C. Wang, P. Zhao, and Q. Chen, "Queue-aware reliable embedding algorithm for 5G network slicing," *Comput. Netw.*, vol. 146, no. 9, pp. 138–150, Dec. 2018.

[9] Y. Liu, Y. Lu, W. Qiao, and X. Chen, "Reliability-aware service chaining mapping in NFV-enabled networks," *ETRI J.*, vol. 41, no. 2, pp. 207–223, Jun. 2019.

[10] L. Qu, C. Assi, K. Shaban, and M. Khabbaz, "Reliability-aware service provisioning in NFV-enabled enterprise datacenter networks," in *Proc. 12th Int. Conf. Netw. Service Manage. (CNSM)*, Montreal, QC, Canada, Oct. 2016, pp. 153–159.

[11] L. Qu, M. Khabbaz, and C. Assi, "Reliability-aware service chaining in carrier-grade softwarized networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 558–573, Mar. 2018.

[12] L. Qu, C. Assi, M. J. Khabbaz, and Y. Ye, "Reliability-aware service function chaining with function decomposition and multipath routing," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 2, pp. 835–848, Jun. 2020.

[13] D. Zhao, J. Ren, R. Lin, S. Xu, and V. Chang, "On orchestrating service function chains in 5G mobile network," *IEEE Access*, vol. 7, pp. 39402–39416, Jan. 2019.

[14] D. Li, P. Hong, K. Xue, and J. Pei, "Virtual network function placement considering resource optimization and SFC requests in cloud datacenter," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 7, pp. 1664–1677, Jul. 2018.

[15] L. Qu, C. Assi, K. Shaban, and M. J. Khabbaz, "A reliability-aware network service chain provisioning with delay guarantees in NFV-enabled enterprise datacenter networks," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 3, pp. 554–568, Sep. 2017.

[16] M. Wang, B. Cheng, and J. Chen, "Joint availability guarantee and resource optimization of virtual network function placement in data center networks," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 2, pp. 821–834, Jun. 2020.

[17] X. Cheng, Y. Wu, G. Min, and A. Y. Zomaya, "Network function virtualization in dynamic networks: A stochastic perspective," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2218–2232, Oct. 2018.

[18] B. Rashidi, C. Fung, and E. Bertino, "A collaborative DDoS defence framework using network function virtualization," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 10, pp. 2483–2497, Oct. 2017.

[19] C. Pham, N. H. Tran, S. Ren, W. Saad, and C. S. Hong, "Traffic-aware and energy-efficient vNF placement for service chaining: Joint sampling and matching approach," *IEEE Trans. Services Comput.*, vol. 13, no. 1, pp. 172–185, Jan. 2020.

[20] M. A. Tahmasbi Nejad, S. Parsaeefard, M. A. Maddah-Ali, T. Mahmoodi, and B. H. Khalaj, "VSPACE: VNF simultaneous placement, admission control and embedding," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 542–557, Mar. 2018.

[21] V. Eramo, E. Miucci, M. Ammar, and F. G. Lavacca, "An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2008–2025, Aug. 2017.

[22] M. A. Raayatpanah and T. Weise, "Virtual network function placement for service function chaining with minimum energy consumption," in *Proc. IEEE Int. Conf. Comput. Commun. Eng. Technol. (CCET)*, Beijing, China, May/Aug. 2018, pp. 198–202.

[23] J. Liu, Z. Jiang, N. Kato, O. Akashi, and A. Takahara, "Reliability evaluation for NFV deployment of future mobile broadband networks," *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 90–96, Jun. 2016.

[24] J. Fan, M. Jiang, and C. Qiao, "Carrier-grade availability-aware mapping of service function chains with on-site backups," in *Proc. IEEE/ACM 25th Int. Symp. Qual. Service (IWQoS)*, Vilanova i la Geltru, Spain, Jun. 2017, pp. 1–10.

[25] J. Fan, M. Jiang, O. Rottenstreich, Y. Zhao, T. Guan, R. Ramesh, S. Das, and C. Qiao, "A framework for provisioning availability of NFV in data center networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2246–2259, Oct. 2018.

[26] N.-T. Dinh and Y. Kim, "An efficient reliability guaranteed deployment scheme for service function chains," *IEEE Access*, vol. 7, pp. 46491–46505, Apr. 2019.

[27] S. Aidi, M. F. Zhani, and Y. Elkhatib, "On improving service chains survivability through efficient backup provisioning," in *Proc. Int. Conf. Netw. Service Manage. (CNSM)*, Rome, Italy, Dec. 2018, pp. 5–9.

[28] M. Karimzadeh-Farshbafan, V. Shah-Mansouri, and D. Niyato, "Reliability aware service placement using a viterbi-based algorithm," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 1, pp. 622–636, Mar. 2020.

[29] Y. Xu and V. P. Kafle, "An availability-enhanced service function chain placement scheme in network function virtualization," *J. Sensor Actuator Netw.*, vol. 8, no. 2, pp. 18–34, Feb. 2019.

[30] J. Sun, G. Zhu, G. Sun, D. Liao, Y. Li, A. K. Sangaiah, M. Ramachandran, and V. Chang, "A reliability-aware approach for resource efficient virtual network function deployment," *IEEE Access*, vol. 6, pp. 18238–18250, Mar. 2018.

[31] J. Zhang, Z. Wang, C. Peng, L. Zhang, T. Huang, and Y. Liu, "RABA: Resource-aware backup allocation for a chain of virtual network functions," in *Proc. IEEE Conf. Comput. Commun.*, Paris, France, Apr. 2019, pp. 1918–1926.

[32] S. Aidi, M. F. Zhani, and Y. Elkhatib, "On optimizing backup sharing through efficient VNF migration," in *Proc. IEEE Conf. Netw. Softw. (NetSoft)*, Paris, France, Jun. 2019, pp. 24–28.
[33] S. Xu, B. Liao, B. Hu, C. Han, C. Yang, Z. Wang, and A. Xiong, "A reliability-and-energy-balanced service function chain mapping and migration method for Internet of Things," *IEEE Access*, vol. 8, pp. 168196–168209, 2020.
[34] J. Fan, Z. Ye, C. Guan, X. Gao, K. Ren, and C. Qiao, "GREP: Guaranteeing reliability with enhanced protection in NFV," in *Proc. ACM SIGCOMM Workshop Hot Topics Middleboxes Netw. Function Virtualization*, London, U.K., Aug. 2015, pp. 13–18.
[35] J. Fan, C. Guan, Y. Zhao, and C. Qiao, "Availability-aware mapping of service function chains," in *Proc. IEEE Conf. Comput. Commun.*, Piscataway, NJ, USA, May 2017, pp. 1–4.
[36] S. Herker, X. An, W. Kiess, S. Beker, and A. Kirstädter, "Datacenter architecture impacts on virtualized network functions service chain embedding with high availability requirements," in *Proc. IEEE Globecom Workshops*, San Diego, CA, USA, Dec. 2015, pp. 1–7.
[37] M. Karimzadeh-Farshbafan, V. Shah-Mansouri, and D. Niyato, "A dynamic reliability-aware service placement for network function virtualization (NFV)," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 2, pp. 318–333, Feb. 2020.
[38] J. Pei, P. Hong, K. Xue, and D. Li, "Efficiently embedding service function chains with dynamic virtual network function placement in geo-distributed cloud system," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 10, pp. 2179–2192, Oct. 2019.
[39] X. Liu and B. Wang, "An algorithm for fragment-aware virtual network reconfiguration," *PLoS ONE*, vol. 13, no. 11, pp. 1–16, Nov. 2018.
[40] X. Liu, B. Wang, and Z. Yang, "Virtual network embedding based on topology potential," *Entropy*, vol. 20, no. 12, pp. 941–954, Dec. 2018.
[41] S. Gong, J. Chen, S. Zhao, and Q. Zhu, "Virtual network embedding with Multi-attribute node ranking based on topsis," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 2, pp. 522–541, Oct. 2016.
[42] S.-Q. Gong, J. Chen, Q.-Y. Kang, Q.-W. Meng, Q.-C. Zhu, and S.-Y. Zhao, "An efficient and coordinated mapping algorithm in virtualized SDN networks," *Frontiers Inf. Technol. Electron. Eng.*, vol. 17, no. 7, pp. 701–716, Jul. 2016.
[43] X. Han, X. Meng, Q. Kang, and Y. Su, "Survivable virtual network link shared protection method based on maximum spanning tree," *IEEE Access*, vol. 7, pp. 92137–92150, Jul. 2019.
[44] Y. Su, X. Meng, Q. Kang, and X. Han, "Survivable virtual network link protection method based on network coding and protection circuit," *IEEE Access*, vol. 6, pp. 67477–67493, Oct. 2018.
[45] X. Li, Z. Yu, Z. Li, and N. Wu, "Group consensus via pinning control for a class of heterogeneous multi-agent systems with input constraints," *Inf. Sci.*, vol. 542, pp. 247–262, Jan. 2021.

DONG ZHAI received the B.S. degree from Shanxi University, Taiyuan, China, in 2016, and the M.S. degree from Air Force Engineering University, Xi'an, China, in 2018. His research interests include network function virtualization and network security.

XIANGRU MENG received the B.S., M.S., and Ph.D. degrees from Xi'an Jiaotong University, China, in 1985, 1988, and 1994, respectively. He is currently a Professor with Air Force Engineering University, Xi'an, China. From 1995 to 1997, he was a Visiting Scholar with the University of Electronic Science and Technology, Chengdu, China. His research interests include the next-generation Internet, network virtualization, and survivable networks.

ZHENHUA YU (Member, IEEE) received the B.S. and M.S. degrees from Xidian University, Xi'an, China, in 1999 and 2003, respectively, and the Ph.D. degree from Xi'an Jiaotong University, Xi'an, in 2006. He is currently a Professor with the Institute of System Security and Control, College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an. He has authored more than 20 technical articles for conferences and journals. He holds two invention patents. His research interests include cyber-physical systems and system security.

XIAOYANG HAN received the B.S. and M.S. degrees from Air Force Engineering University, China, in 2009 and 2017, respectively, where he is currently pursuing the Ph.D. degree. His research interests include survivable virtual networks and software defined networks.